

Towards Player Adaptivity in a Serious Game for Conflict Resolution

Corrado Grappiolo, Yun-Gyung Cheong, Julian Togelius, Rilla Khaled, Georgios N. Yannakakis
Centre for Computer Games Research
IT University of Copenhagen
Rued Langgaards Vej 7 Copenhagen, Denmark
Email: {cogr, yugc, juto, rikh, yannakakis}@itu.dk

Abstract—We present a technology demonstrator for an adaptive serious game for teaching conflict resolution and discuss the research questions associated with the project. The prototype is a single-player 3D mini-game which simulates a resource management conflict scenario. In order to teach the player how to resolve this type of conflict, the underlying system generates level content automatically which adapts to player experience and behaviour. Preliminary results demonstrate the efficiency of the procedural content generation mechanism in guiding the training of players towards targeted learning objectives.

Index Terms—Serious Games Technology, Procedural Content Generation, Player Modelling.

I. INTRODUCTION

Conflict has always been a problem in society. As society has continued to evolve, for example, as patterns of migration change, complex conflicts have seemingly become more commonplace. Social interaction and integration, both within and between cultures, represent one of the main sources of conflict generation; hence conflict management and resolution are fundamentally important skills to be teaching the upcoming generation. In particular, it is important for young children to learn how to deal with conflicts and resolve them in the absence of external mediation. But teaching such skills can be tedious and non-trivial. Already, there exist a number of pedagogical styles for teaching conflict resolution in school classes, such as peer mediation training (where children are trained how to resolve conflicts amongst themselves), direct teaching (where children are taught via lessons, films on DVD, readings) and drama workshops [1] (which provide an immersive and free environment from which to explore conflict resolution). We aim to show that conflict resolution skills can be effectively taught to school kids using game-based teaching methods.

A. Serious games

Game technology in the form of serious games has already proven viable and effective for supporting therapy [2], promoting intercultural communication [3], increasing understanding of ethnic, religious and historical funded conflicts [4], and representing different perspectives on issues such as global politics and foreign policy [5].

Games are promising as learning vehicles for a number of reasons. For example, they afford special kinds of immersion, that cannot be matched by reading a book or watching a film [6]. Related to immersion in games is agency: players can interactively take actions and make decisions that impact on game content and progress. Games allow players to role play in ways similar to drama workshops: within bounded, virtual environments, they can adopt new perspectives and play different roles [7]. But whereas in a drama workshop, the instructor has to interrupt an enactment when the dynamics of role play deviate significantly from the intended plot, with digital games, instead, the ongoing narrative can be adjusted in ways hidden from players [8].

Games also facilitate the experiencing of multiple perspectives on one issue or situation. For example, in the context of conflict resolution, a player could first play the role of aggressor, then later the role of mediator, each time applying different conflict resolution strategies. This enables flexibility and reuse in terms of exploring strategies, as players get to experiment with resolving different conflict scenarios in different ways. Finally, given the sensitive nature of conflict, access to a safe, consequence-free, virtual environment in which to explore behaviours is one of the most important contributions of a conflict resolution game from a pedagogical perspective [7], [9]. Such a game could conceal the true identity of players behind the avatars; in this way the children will be less inhibited in performing actions which they would never perform if their identities were known. For example, a passive and withdrawn child who would normally be afraid of asserting himself towards an aggressive child has the opportunity to play a completely different role in a virtual environment where identities are kept secret.

B. Related Work

There are several examples of intelligent tutoring systems (see [10], [11], [12] among others) which may be classified under the category of educational games, though they do not usually focus on soft social skills, such as conflict resolution. 80 Days [13] is a promising research project of narrative-based educational game with adaptive behaviour. It is an ongoing project in which the player helps an alien, Feon, to write a travel guide about planet Earth. The 80 Days system contains two components: the interactive storytelling component and

the adaptation component. The interactive storytelling component is used as a means to teach geography to the player, and the adaptation is used to track the player types (killer, achiever, socializer and explorer) and put them in the game context. FearNOT! [14], [15] is a serious game that utilises advanced game technology with the purpose of fostering social skills. In this game, the player plays the role of an invisible friend of a character who is a victim of bullying. The player has to interact with his friend and advise him in order to cope with the character's problems. The underlying technique is the FATiMA emotion-driven architecture [16] which models the NPC actors' actions for a given goal and emotional states. Apart from this study, to the best of the authors' knowledge, no other existing serious game has focused on training conflict resolution exploiting the use of advanced game technology.

C. Our contribution

Motivated by this research and technological gap, we have embarked on the development of an educational multiplayer video game, targeted at students between the ages of 10 - 12. The objective of players will be to face and resolve conflicts of different types. While we have not yet fully defined our typology of conflict, or the full set of learning outcomes we hope to encourage in players, we have made progress in establishing key technological components of the game and their interactions. The finished game will be heavily based on current research, and feature several technologies which have not yet been combined in the context of serious games, in particular natural interaction, user modelling, procedural content generation and interactive storytelling [17].

This paper presents our early progress in realising some of the aforementioned game technology components, their interaction, and mapping them to conflict-related learning objectives. Currently, our system consists of a 3D single-player mini-game which simulates a resource management (RM) conflict scenario. In this game, the player has to distribute scarce resources to a number of NPCs, with the goal of keeping all of the NPCs happy. The game learns the playing style of the player, in particular to what extent he or she distributes the resources fairly and cooperatively. Every time the player finishes a level, the game will generate a new level algorithmically designed, based on the player's experience and to the game's own prediction, in order to lead the player toward maximum levels of fairness and cooperation.

The technology demonstrator incorporates two interconnected components: a player modeling module and a procedural content generation module. We have adopted a modular game engine design to facilitate future enrichment of the existing components, and extensions to the architecture, which will be necessitated as our research progresses.

Before delving into the details of the mini-game in Section III, we first provide a general description of the components present in our mini-game and position them within the literature (see Section II). Section IV presents our pilot study. Current research outcomes are discussed in Section V followed by the paper's main conclusions in Section VI.

II. ADAPTATION SCHEME FOR CONFLICT RESOLUTION

The optimal strategy for conflict resolution is not apparent and certainly not unique; strategies for resolving conflicts usually depend on the group dynamics, the cultural backgrounds and the emotional states of the conflicting subjects (i.e. players) and the ability of the involved subjects to perceive the existence of a conflict. All these challenges justify the need of Artificial Intelligence (AI) tools, as middleware solutions within the game architecture, which can automatically identify such player variability and game uncertainty and, further, personalise the training for conflict resolution. Thus, the serious game we envision aims at generating conflict scenarios in real time which would adapt to the player behaviour and drive her toward specific conflict resolution outcomes. Adaptation is needed because we aim to create a tool for implicit learning: the game should provide scenarios which fit with the player behaviours, personalities, and learning capabilities.

For this study we opt for adaptive game content based on computational player models [18]. Any game element may have an impact on player experience including objects in the 3D world, scenario typology, and features which characterise a particular conflict. The top-level of the adaptation mechanism, sketched in Fig. 1, is synthesised by two main components: the game engine and the AI middleware. The AI middleware is composed of a procedural content generation (PCG) module and a player modeling (PM) module. Our adaptation scheme first generates game content, representing a level of the game (arrow 1). Then, the user plays the generated content (illustrated by arrow 2 in the figure). Once the level is terminated, the game engine returns data regarding the gameplay to the AI middleware (arrow 3) which updates the player model module. The updated PM module informs the PCG module and drives the generation of the next level. More specifically, an evaluation criterion (based on the PM, arrow 4) assess the goodness (fitness) of a population of candidate levels (arrow 5) the most fit of which is provided to the game engine as the next level (arrow 1).

III. THE RESOURCE MANAGEMENT MINI-GAME

The mini-game presented in this paper is the embryo of the final conflict resolution game and the initial study towards meeting the objectives set in Section II. The conflict scenario simulated in this game is a simple Resource Management (RM) scenario. The mini-game is single-player with a third-person perspective, developed on the Unity3D game engine¹ and written in C#. It runs as a stand-alone application on MacOS machines. The player controls the main character (see Fig. 2) and finds herself in a physical environment populated by Moai figures² and resources represented as fireballs. The environment, although continuous, is discretised in cells: each level is composed of $w \times h$ cells, where w and h is the width and height of the level, respectively.

¹<http://www.unity3d.com>

²<http://en.wikipedia.org/moai>

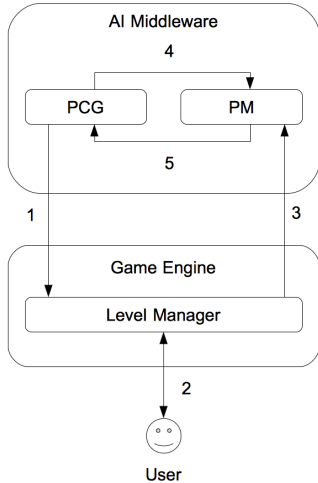


Fig. 1. The top-level architecture of the conflict resolution game. The arrows represent the procedural steps which realise the interaction between the AI middleware, the game engine and the user.

The Moais have a *happiness* value, which decreases over time. Graphically, the less happy the Moais are, the more transparent their hats get. The only way to get happier is to acquire fireballs present in the environment. The fireballs are associated with different levels of happiness that they can elicit in the Moais: the bigger they are, the higher the Moai happiness. However, the Moais cannot actively collect resources; it is up to the player to distribute those fireballs. Moais are divided into two groups, represented as blue and red colors. The game is divided into 10 levels of variable duration, ranging from 30 seconds to 180 seconds. At the end of each level the game provides a reward to the player, according to the strategy she adopted in that level, in the form of emoticons (very sad, sad, serious and happy faces). The goal of the player is to collect happy faces. Figure 2 is a screenshot of the game. The current game setting is designed for a single-player, but could easily be extended to a multiplayer environment at a later development cycle. This allows us, for instance, to focus our attention on the player modelling component first and then concentrate on its relationship with group modelling.

A. Conflict Management Strategies

For effective conflict resolution, our player modelling is based on the work of Thomas [19] on conflict management strategies. Thomas suggests a taxonomy for conflict-handling modes based on two dimensions: cooperativeness and assertiveness. The former is related to the attempt of satisfying other's concerns, whereas the latter specifies the attempt to satisfying one's own concerns. Those modes are: *Competition*, *Collaboration*, *Compromise*, *Avoiding* and *Accommodation*. Figure 3 illustrates how these conflict-handling modes are represented within the two aforementioned dimensions.

An *avoiding* behaviour occurs when the player does not face the conflict. In our mini-game this happens when the player does not distribute the resources and, instead, she just

wanders around. A *competitive* behaviour within the game could be mapped to a player who provides the fireballs to one Moai type only. This strategy would result in all red Moais being happy and all blue Moais being unhappy. On the other hand, a *collaborative* (cooperative) strategy is simulated when the player distributes resources in such a way that the happiness levels between Moais are even. Any other mixed strategies of resource distributions are interpreted as *compromising* strategies. Finally, a player who first makes happy one Moai type and then the other type can be considered adopting an *accommodating* strategy. For a matter of simplicity for our RM scenario we treat accommodation strategy under the compromising strategy.

The learning objective of the RM scenario is to be cooperative. It is designed to be taught implicitly through the gameplay. For that purpose we adopt the reward signal and policy learning principles used in Reinforcement Learning (RL) [20]. The idea is to provide a feedback (reward) to the player, in form of happy faces (positive reward), serious (neutral reward), sad and very sad faces (negative and very negative reward), at the end of each level, depending on the player's behaviour during the last level played. Assuming that the player would be willing to maximise the number of smiles she gets, players are expected to learn the optimal policy (i.e. cooperation) for conflict resolution eventually.

B. Player Modelling

We require the RM mini-game to generate personalised playing experiences and drive the players toward player-centered learning objectives. It is therefore essential to construct computational models of players (see [21], [22], [23] among others) that would be able to recognise dissimilarities in playing style and preferences.

The player modelling component of the RM mini-game focuses on quantifying the degree up to which the player



Fig. 2. A screenshot of a level of the Resource Management Mini-Game. The player character, one resource item and several Moais are visible.

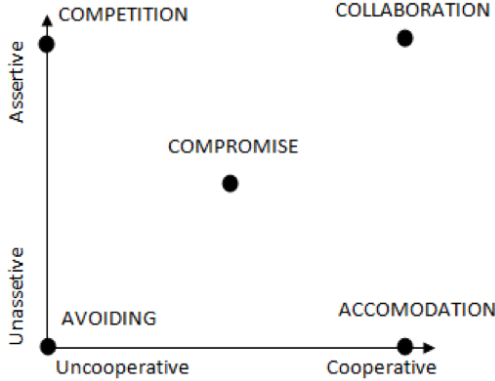


Fig. 3. Conflict-handling modes.

follows a collaborative strategy when playing a specific level. In particular, PM is based on player actions (picking and distributing fireballs) and game-state transitions during the level played to detect the conflict strategy the player has adopted. To calculate the actual cooperation value we take into account how the player distributes the fireballs throughout the whole level. The average happiness values of the two Moai types are calculated each time a fireball is distributed and the standard deviation of these values is calculated. Specifically, the actual level of cooperation, $C_{a,t}$, at the end of level number t ($t \in [1, 10]$) in the order of levels played, is obtained via the following heuristic:

$$C_{a,t} = 1 - 2\sigma_{RB} \quad (1)$$

where σ_{RB} is the standard deviation of the two average happiness values, h_R (red) and h_B (blue), of the Moais present in the game environment. The σ_{RB} value provides information about the difference between h_R and h_B across time, thus, the lower σ_{RB} is the higher, we assume, indicates the collaboration strategy in this game. Given that the happiness values of the Moais lie within $[0, 1]$ the σ_{RB} value lies within the $[0, 0.5]$ interval. The $C_{a,t}$ value is normalised to $[0, 1]$ by multiplying σ_{RB} by 2.

The actual cooperation level of a player can only be determined at the end of each level. As mentioned earlier (see Fig 1), the goodness of a level in terms of cooperation is *predicted*. The difference between the actual and the predicted cooperation level introduces an error that the player model attempts to minimise, thereby, approximating the level of actual cooperation for each player at a specific level.

Although the mini-game adopts a simple scenario and the data set recorded during gameplay is small, it is still difficult to ad hoc design a mathematical function mapping human behaviours which are likely non-linear. The universal approximation capability of Artificial Neural Networks (ANNs) makes them appropriate for the underlying player modelling task. In this paper ANNs attempt to approximate the unknown function between game level elements and player cooperation. That

function is essential when various levels need to be simulated, evaluated and optimised to maximise cooperation personalise the game levels and the conflict management strategy for a particular player.

The ANN outputs a single value that represents the predicted level of collaboration (C_p). On the other hand, the ANN input vector contains the following nine gameplay statistical features: width and height of the level (2 features); duration of the level in seconds (1 feature); number of red Moais, blue Moais and fireballs (3 features); and initial average happiness of red Moais, blue Moais, and fireball size (3 features). All features are uniformly normalised in the $[0, 1]$ interval before they are presented to the network. A single-hidden layer ANN architecture with 11 hidden neurons — each employing the logistic (sigmoid) activation function — is implemented in this paper. This topology is selected after extensive experimentation with several single-hidden layer architectures.

The actual cooperation values $C_{a,t}$ obtained via (1) are used to train the network via non-batch backpropagation [24] at the end of each level played. The number of epochs is 50 for real-time training. Due to the small number of epochs, we employed a high learning rate value, 0.9.

The reward provided at the end of the level is dependent on the player strategy: if the player's actual cooperation at level number t has an increasing trend then a happy face is presented as a reward; a sad face is presented when the player's actual cooperation is decreasing; otherwise, if the actual cooperation value is not changing notably (less than 5% in this paper) a serious face is presented as a reward.

C. Experience-driven PCG

For the experience-driven PCG (EDPCG) [18] component of the RM mini-game we employ a search-based approach. Search-based PCG (SBPCG) [25] is a technique which mainly relies on Evolutionary Computation (EC) techniques in which a population of candidate individuals (solutions) representing content are selected based on an evaluation (fitness) function. Selected solutions are decoded and provided as content to the user for playing. Following Darwin's theory of evolution, an evolutionary process discards poorly-fit individuals and replaces them with new offspring, generated by the recombination of highly fit individuals (crossover) and/or small stochastic variations of the individual's genes (mutation) generating new offspring in the population.

There are a few successful examples of EDPCG in the literature that have inspired the design of this component. Shaker et al. [26] propose an adaptive mechanism that automatically designs Super Mario levels so that player's modelled fun is maximised. Fun (and other dimensions of player experience) is modelled via in-game player data and self-reported preferences; the models are constructed using neuro-evolutionary preference learning [27], [28]. Other aspects of games that have been considered for PCG include environments, such as tracks for racing games [29], levels for platform games [30], [31], narrative [32], [33], rules for board games [34], [35] and Pac-Man-like games [36].

The EDPCG component of the resource management mini-game is based on a Genetic Algorithm (GA) implementation. The content generated by EDPCG designs a new level for the player. According to the algorithm followed, each level (member of the GA population) is represented by a string of bits which encode the following information:

- 1) level size (width, height), 3 bits each;
- 2) time available (in seconds), 8 bits;
- 3) number of red and blue Moais, and number of fireballs, 3 bits each;
- 4) position (6 bits) and initial happiness of each Moai (4 bits) and fireball size (4 bits).

The genome has an overall length of 173 bits. The chromosome size is fixed: it defines the properties for all the Moais and Fireballs which could be present in the environment (point 4) but only the first of them, specified by point 3, are taken into consideration. While alternative representations are possible the bit-string encoding of the GA was adopted due to its simplicity.

The design of the fitness function is based on the PM component which, in turn, is built on the behavioural responses of the player. The EDPCG algorithm searches for levels that minimise the following fitness function:

$$f(x) = \left| C_p(x) - \frac{1}{2}(C_{a,t-1} + 1) \right| \quad (2)$$

where $f(x)$ is the fitness function of the x individual, $C_p(x)$ is the predicted cooperation (ANN output) of the player if she would play the level-individual x and $C_{a,t-1}$ is the actual cooperation value the player had at the previous level played (computed by (1)).

The rationale behind the design of the fitness function is as follows. At time t we need to select an individual x among the candidate individuals which leads the player toward the learning goal (maximum collaboration, the constant value 1 in the formula), before she plays it: $C_p(x)$, the predicted level of cooperation for individual x . However, we do not want to simply select individuals such that $f(x) = C_p(x) = 1$. This formula does not consider what the player has been doing so far; in other words, it does not adapt to player behaviour. As an example, let us assume at time $t-1$ we provide a level such that $f(x) = C_p(x) = 1$. The level is very difficult and thus the player's actual cooperation value is decreasing. If the next level (at time t) targets the cooperation value of $f(x) = C_p(x) = 1$ again, this may lead to low performance. Instead, at time t , we set the target cooperation value as $\frac{1}{2}(C_{a,t-1} + 1)$, taking the middle of the maximum cooperation value (i.e., 1) and the player's cooperation value at the previous level, $C_{a,t-1}$. In this way, we aim to bring the player toward maximum collaboration (fairness in the resource distribution task) in a gradual way, which takes into consideration the player's behaviour and adapts to it.

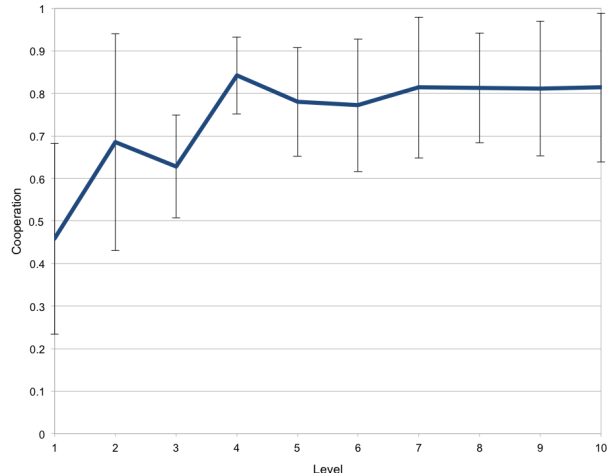


Fig. 4. The evolution of the average actual cooperation value C_a and its corresponding standard deviation across all six experiment subjects during a gameplay (10 levels).

IV. A PILOT STUDY

We designed a pilot study to test if the game adapts to the player's behavior and the capability to drive the player toward the envisaged learning goal. A total of 6 subjects (one male and five females) recruited from IT University of Copenhagen and social communities voluntarily participated in this study. Their age ranged from 19 to 58. We are aware that the targeted age for the final Conflict Resolution game is very different (10-12 years of age); the aim of the experiment was to gather preliminary results which could validate the efficiency of the adaptation mechanism.

The subjects were asked to familiarise with the user interface and the gameplay. Then, the study operator explained the concept of happiness, the way the Moais can get happy (distributing resources), their goal (obtaining as many smiles as possible) and the strong assumption regarding the resource distribution (they must distribute all or as many resources as they can, providing the time available).

The game is composed of 10 levels in total. The first level was fixed for all the participants: in a 2×2 environment consisting of one fireball, one red and one blue Moai, and the time constraint of 30 seconds. The level content was automatically generated from the second level up to the tenth level. On completion, the first level returns an outcome indicating if the player is avoiding the conflict or adopting a competitive strategy.

The average actual cooperation level $C_{a,t}$ and its standard deviation across all six subjects, from the first to the last level, is shown in Fig. 4. For the last two levels, however, the statistic is based on five participants only because one participant was unable to complete those levels. The graph shows that the average level of player cooperation increased as the game progresses. In particular, the actual cooperation

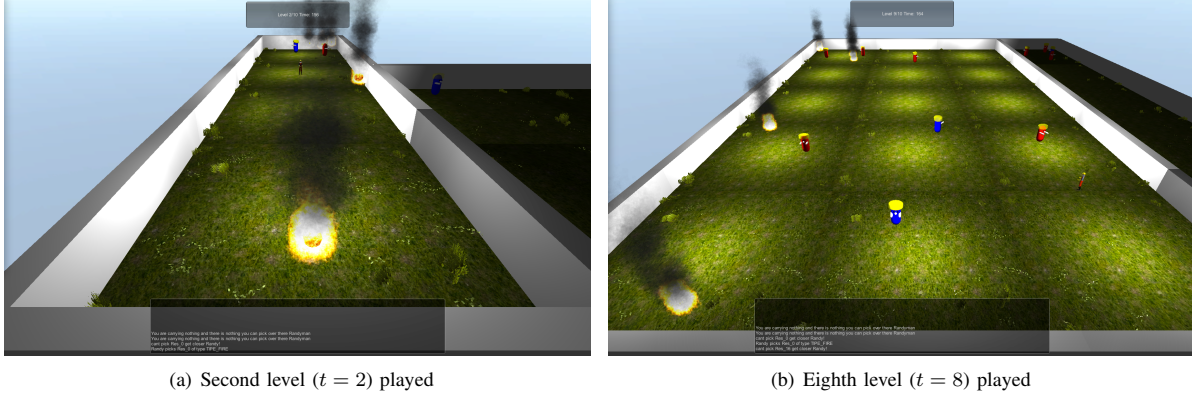


Fig. 5. Example levels generated for subject no. 6.

value of the last level played (≈ 0.81) is much higher than the corresponding value of the first level (≈ 0.46). A two-tailed t-test was conducted on the pair of cooperation values indicated that the difference was significant with a 95% of confidence ($t(8) = 2.75$, p value < 0.05). the standard deviation of the cooperation value decreases as the game progresses (from ≈ 0.22 to ≈ 0.18).

Fig. 5(a) shows a second level scene for a particular subject, of which setting was generated by the PCG after the PM component has been updated with the data collected from the first level gameplay. The second level contained one red (happiness 0.2), one blue Moai (happiness 0.9) and five Fireballs (average happiness 0.38) within the time limit of 180 seconds. The red Moai was sadder than the blue one as the consequence of the player behavior at the previous level; she gave the unique Fireball to the blue Moai (her actual cooperation value at the end of level one was $C_{a,1} = 0.5$). The five Fireballs' happiness distribution was (0.2, 0.7, 0.5, 0.2, 0.2). The level had the size of 4×1 . The player's distribution sequence and Fireball happinesses was 0.5 to red, 0.2 to blue, and all the remaining to red. At the end of this level, the player's actual cooperation was $C_{a,2} = 0.8$. For this strategy the player received a smiley face.

Fig. 5(b) shows a scene at the eighth level of which size was 3×5 . It contained 5 red, 2 blue Moais, 5 Fireballs with the happiness distribution of (0.2, 0.3, 0.6, 0.5, 0.2), and the time limit of 180 seconds. The average happiness for red Moais was 0.67 and the average happiness for the blue Moais was 0.82. The player managed to distribute all the resources except for one. The distributed Fireballs were given to red Moais, which was not an optimal strategy in terms of cooperation; the player first delivered Fireballs with happiness 0.3 and 0.2, reducing σ_{RB} ; she then delivered the Fireball with happiness 0.6, increasing σ_{RB} from 0.03 to 0.04. At this point the happiness among the Moais was close to even. Then the player delivered the Fireball with happiness 0.5 to one red Moai; σ_{RB} became 0.08 which was the same value as σ_{RG} at the beginning of the level. The time ran over when one Fireball still remained. A serious face was provided as reward.

V. DISCUSSION

The preliminary experimental results has shown the feasibility of our approach to combine the player modeling (PM) and the Experience-driven Procedural Content Generation (ED-PCG) approaches for generating adaptive conflict generation game contents. However, the following issues still remain:

- The cooperation formula (1) calculates the level of fair distribution across different types of Moais based on a single-player's performance. An ideal conflict generation and cooperation setting, however, requires multiple players. For instance, a person A is cooperative when he helps B do her homework. Or, A and B in a same team are cooperative in achieving a mission. The pilot study presented in this paper aimed to evaluate the adaptation functionality of the system rather than evaluating the simulation setting for conflict generation and management.
- The resource management game in the pilot study drove the players towards high level of cooperation, assuming that maximum cooperativeness is always the best solution. However, in a more complicated setting different management strategies can be employed depending on situation. For instance, a competitive or ignoring strategy can be used to respond to a competitive counterpart when the cooperative problem-solving strategy is not available.
- The escalating pattern of cooperation in the pilot study can be explained in two directions: player adaptation and play improvement. The first rationale is that the system correctly adjusted the level setting according to the gamer's play at the previous level. Secondly, the players learned the game mechanics (fair distribution in our case) as they went through multiple game levels. Since the reasoning behind the increasing pattern is not clear, we conducted a preliminary study to compare our result with a baseline setting where a random level generator was used. The average of 5 participants' gameplay has shown an oscillating pattern in which high and low cooperation values are alternated as level increases. Although this pattern has indicated that our

procedural content generation component is effective in player adaptation, the result is not statistically significant to support the connection. A formal evaluation will be needed to measure the adaptivity and play improvement separately.

VI. CONCLUSIONS AND FUTURE WORK

This paper presented a metaheuristic-search procedural content generation approach in a simple resource management game. The content generator is driven by a neural network predictor of the distribution fairness of the player. Our pilot study shown that the generated content drove the players towards a specific learning target. To address the limitations discovered in the pilot study, we plan to conduct an extensive game user study in order to test the validity of the cooperation value and the effectiveness of the player modelling component. In addition we plan to implement the complete system in a multi-player game environment to simulate social interaction that will foster conflict resolution skills.

ACKNOWLEDGMENT

This work has been supported, in part, by the FP7 ICT project SIREN (project no: 258453).

REFERENCES

- [1] Catterall and S. James, "Enhancing peer conflict resolution skills through drama: an experimental study," *Research in Drama Education*, vol. 12, no. 2, pp. 163–178, Jun. 2007.
- [2] J. DiFede and H. Hoffman, "Virtual Reality Exposure Therapy for World Trade Center Post-traumatic Stress Disorder: A Case Report," in *Cyber Psychology and Behaviour*, vol. 5, no. 6, 2002, pp. 529–535.
- [3] E. Raybourn, "Computer game design: New directions for intercultural simulation game designers," in *In Developments in Business Simulation and Experiential Exercises*, no. 24, 1997.
- [4] T. Buch and S. Egenfeldt-Nielsen, "The Learning Effects of Global Conflicts: Palestine," in *Media@Terra Conference Proceedings*, 2006.
- [5] G. Frasca, "Ideological Videogames: Press left button to dissent," in *International Games Developers Association, Ivory Tower*, November 2003.
- [6] G. Calleja, "In-Game: From Immersion to Incorporation," in *MIT Press*, 2011.
- [7] K. Squir, "From Content to Context: Videogames as Designed Experience," in *Educational Researcher*, vol. 35, no. 8, November 2006, pp. 19–29.
- [8] C. J. Saretto and R. M. Young, "Mediation in Mimesis Liquid Narratives," in *Proceedings of ACSME 2001*, 2001.
- [9] P. Pedersen, "Simulations: A Safe Place to Take Risks in Discussing Cultural Differences," in *Simulation & Gaming*, vol. 26, no. 2, June 1995, pp. 201–06.
- [10] D. S. McNamara, G. T. Jackson, and A. Graesser, "Intelligent tutoring and games (ITaG)," in *AIED Workshops Proceedings - Intelligent Educational Games*, vol. 3, 2009.
- [11] J. A. Walonoski and N. T. Heffernan, "Detection and Analysis of Off-Task Gaming Behavior in Intelligent Tutoring Systems," in *Lecture Notes in Computer Science*, 2006, Volume 4053/2006, 382–391, 2006.
- [12] J. Rowe, L. Shores, B. Mott, and J. Lester, "Integrating Learning and Engagement in Narrative-Centered Learning Environments," in *In Proceedings of the Tenth International Conference on Intelligent Tutoring Systems (ITS-10)*, Pittsburgh, Pennsylvania, pp. 166–177, 2010.
- [13] S. Göbel, V. Wendel, C. Ritter, and R. Steinmetz, "Personalized, adaptive digital educational games using narrative game-based learning objects," in *Proceedings of the Entertainment for education, and 5th international conference on E-learning and games, ser. Edutainment'10*. Berlin, Heidelberg: Springer-Verlag, 2010, pp. 438–445.
- [14] R. Aylett, M. Vala, P. Sequeira, and A. Paiva, "FearNot! an emergent narrative approach to virtual dramas for anti-bullying education," in *International Conference on Virtual Storytelling, Saint Malo, France*, 2007.
- [15] M. Vala, P. Sequeira, A. Paiva, and R. Aylett, "FearNot! Demo - A Virtual Environment with Synthetic Characters to help Bullying," in *The Sixth Intl. Joint Conf. on Autonomous Agents and Multi-Agent Systems AAMAS07, Honolulu, Hawaii, USA.*, May 2007.
- [16] J. Dias and A. Paiva, "Feeling and Reasoning: a Computational Model for Emotional Agents," in *Intelligent Agents and Synthetic Characters Group*, 2005.
- [17] G. N. Yannakakis, J. Togelius, R. Khaled, A. Jhala, K. Karpouzis, A. Paiva, and A. Vasalou, "Siren: Towards Adaptive Serious Games for Teaching Conflict Resolution," in *Proceedings European Conference on Games-Based Learning (ECGBL)*, Copenhagen, 2010, pp. 412–417.
- [18] G. N. Yannakakis and J. Togelius, "Experience-Driven Procedural Content Generation," *IEEE Transactions on Affective Computing*, in print 2011.
- [19] K. W. Thomas, "Conflict and conflict management: Reflections and update," in *Journal of Organizational Behavior*, vol. 13, no. 3, 1992, pp. 265–274.
- [20] R. S. Sutton and A. G. Barto, "Reinforcement Learning: An Introduction," in *MIT Press, Cambridge, MA, A Bradford Book*, 1998.
- [21] R. Houlette, "Player Modeling for Adaptive Games. AI Game Programming Wisdom II," in *Charles River Media, Inc.*, 2004, pp. 557–566.
- [22] D. Charles and M. Black, "Dynamic player modelling: A framework for player-centric digital games," in *in Proceedings of the International Conference on Computer Games: Artificial Intelligence, Design and Education*, 2004, pp. 2935, 2004.
- [23] A. Drachen, A. Canossa, and G. N. Yannakakis, "Player Modeling using Self-Organization in Tomb Raider: Underworld," in *Proceedings of the IEEE Symposium on Computational Intelligence and Games (CIG)*, Milan, September 2009.
- [24] R. Rojas, "Neural Networks - A Systematic Introduction," in *Springer-Verlag, Berlin, New York*, vol. 502, 1996.
- [25] J. Togelius, G. N. Yannakakis, K. O. Stanley, and C. Browne, "Search-based procedural content generation," in *Proceedings of the EvoStar Conference*. Springer-Verlag, April 2010.
- [26] N. Shaker, G. Yannakakis, and J. Togelius, "Towards Automatic Personalized Content Generation for Platform Games," in *in Proceedings of the Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE)*, 2010.
- [27] C. Pedersen, J. Togelius, and G. N. Yannakakis, "Modeling player experience for content creation," *IEEE Transactions on Computational Intelligence and AI in Games*, vol. 2, no. 1, pp. 54–67, 2010.
- [28] H. P. Martinez, K. Hullett, and G. N. Yannakakis, "Extending neuro-evolutionary preference learning through player modeling," in *Proc. IEEE Conference on Computational Intelligence and Games (CIG)*, Copenhagen, Denmark, August 2010, pp. 313–320.
- [29] J. Togelius, R. D. Nardi, and S. M. Lucas, "Towards automatic personalised content creation for racing games," in *Proceedings of IEEE Symposium on Computational Intelligence and Games (CIG)*, 2007.
- [30] K. Compton and M. Mateas, "Procedural level design for platform games," in *in Proceedings of the Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE)*, 2006.
- [31] G. Smith, M. Treanor, J. Whitehead, and M. Mateas, "Rhythm-based level generation for 2d platformers," in *in Proceedings Foundations of Digital Games*, 2009.
- [32] M. J. Nelson, C. Ashmore, and M. Mateas, "Authoring an interactive narrative with declarative optimization-based drama management," in *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE)*, 2006.
- [33] M. O. Riedl and N. Sugandh, "Story planning with vignettes: Toward overcoming the content production bottleneck," in *Lecture Notes in Computer Science*, vol. 5334, 2008, pp. 168–179.
- [34] C. Browne, "Automatic generation and evaluation of recombination games," in *Ph.D. dissertation, Queensland University of Technology, Brisbane, Australia*, 2008.
- [35] J. Mark and V. Hom, "Automatic design of balanced board games," in *Proceedings of the Artificial Intelligence and Interactive Digital Entertainment International Conference (AIIDE)*, 2007, pp. 25–30.
- [36] J. Togelius and J. Schmidhuber, "An experiment in automatic game design," in *in Proceedings of the IEEE Symposium on Computational Intelligence and Games*, 2008.